REMARKS

Claims 1-19 were rejected under 35 USC 102 and/or 35 USC 103. Claims 1-13 and 16-19 have been canceled above. Original claims 14 and 15 have been amended, and claims 20-25 have been added above. Applicants respectfully traverse these rejections as applied to the pending claims, based on the following.

Amended claim 14 recites a method for managing a plurality of servers in a cluster. A threshold equal to an integer greater than one is set. A request is sent to one of the servers, and a determination is made that the one server is currently operational but did not successfully handle the request within a predetermined amount of time. In response, a count is incremented and compared to the threshold, and a determination is made that the count is less than the threshold. In response, no corrective action is taken. Another request is sent to the one server, a determination is made that the one server is currently operational but did not successfully handle the request within the predetermined amount of time. In response, the count is incremented and compared to the threshold, and a determination is made that the count equals or exceeds the threshold. In response, a memory dump of the one server is automatically initiated.

The Examiner cited Holt et al. against all but the last element of amended claim 14. The last element of amended claim 14 was originally in claim 17 (canceled above), and the Examiner cited Smullen et al. against original claim 17. Amended claim 14 is a combination of steps to determine when the one server is operational but repeatedly non responsive or very slow to respond, and to automatically initiate a memory dump of the one server as the corrective action. Thus, the present invention as recited in claim 17 is the determination of the proper corrective action, i.e. initiating the memory dump, when the one server is operational but repeatedly non responsive or very slow to respond. Smullen et al. disclose a memory dump, i.e. "to dump the memory before reloading or rebooting the processor", as part of a corrective action when the processor has halted and needs to be reloaded or rebooted:

"the fast memory dump process is triggered by the detection of a CPU halt 100. Upon detection of the CPU halt, the RCVDUMP program will be called 102 with the PRTMEM and RELOAD options. RCVDUMP will then send a START_DUMP call to notify the system 10, 11 that a memory dump is commencing 104." Smullen et al. Column 5 lines 27-33.

Therefore, Smullen et al. do not teach an automatic initiation of a memory dump when a server is operational but repeatedly non responsive or very slow to respond. This would not have been obvious in view of Smullen et al. because Smullen et al. only sense when the processor has halted.

Independent claim 22 distinguishes over Smullen et al. for the same reason that amended claim 14 distinguishes thereover.

Independent claim 20 recites:

> means, responsive to said one server not currently being operational, for automatically issuing a remote restart of said one server,
>
> means, responsive to said one server currently being operational and handling said request after said first predetermined time but before said second predetermined time, for automatically notifying a dispatcher for said one server to reduce a rate of dispatching new requests to said one server, and
>
> means, responsive to said one server currently being operational but not handling said request by said second predetermined time, for automatically initiating a memory dump of said one server.

Thus, new claim 20 recites three different problematic states of the one server, and three different, respective corrective actions best suited for the respective type of problem. This set of corrective actions, tailored to the respective type of problem, was not taught or suggested by the prior art. While Shin teach restarting of a server which does not respond for a predetermined time, Kubo et al. teach load balancing of a slow to respond server, and Smullen et al. teach a memory dump of a processor which has halted, this does not teach all of the elements of claim 20 or the combination of claim 20 recited above.

More specifically, claim 20 recites three progressive levels of failure of the one server and the optimum, automatic corrective action for each level of failure. In the lowest level of failure, the one server is slow to respond (compared to a first predetermined time), and the corrective action is to notify a dispatcher to reduce a rate of dispatching new requests to said one server. In the middle level of failure, the one server is very slow to respond (compared to a second, greater predetermined time), and the corrective action is to automatically initiate a memory dump. In the highest level of failure, the one server is not operational, and the corrective action is to automatically issue a remote restart of the one server. As noted above, the prior art does not teach the memory dump as a response to a server which is operational but very slow to respond. Also, the use of two different predetermined levels in one system to identify slow and very slow responses, and corresponding corrective actions appears new. Also, the prior art does not teach the foregoing combination of three types of problems and three respective types of corrective actions.

Independent claim 24 distinguishes over the prior art for similar reasons that claim 20 distinguishes thereover.

Claim 21 further recites:

> "further comprising a network dispatcher for receiving client requests from client computers and dispatching said client requests to said servers including said one server; and
>
> wherein said first request, said other request and the automatic initiation of the memory dump bypass said network dispatcher."

The prior art does not teach this arrangement either where client requests go through a dispatcher en route to the servers, and the test requests and the automatic initiation of the memory dump bypass the dispatcher en route to the one server. Kubo teach,

> "a terminal monitors the response time of processing at the server, an if the response time exceeds a predetermined value, then the terminal sends a path change request to change the executing computer to another computer." Kubo Column 1 lines 43-46.

Kubo does not mention an automatic initiation of a memory dump or that it bypasses a dispatcher which forwards client requests to the servers. Therefore, claim 21 was not taught or obvious in view of Kubo.

Claims 15, 23 and 25 similarly distinguishes over Kubo.

Based on the foregoing, Applicants request allowance of the present patent application as amended above.

Respectfully submitted,

Arthur J. Samodovitz
Reg. No. 31,297

Dated: March 27, 2006
607-429-4368 (Phone)
607-429-4119 (Fax)